Listing to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Original) Digital data compression encoder, characterized in that it includes:

- an input for a first data flow (S_H), and a second data flow (S_L),

- an encoding module, matching symbols of the first data flow and code words, wherein, for certain symbols, there exist several words, called redundant, corresponding to the same symbol, and

- a processing module for encoding the symbols of the first data flow based on the match, by selecting among the redundant words, on the basis of at least part of the second data flow.

2. (Original) Encoder according to claim 1, characterized in that the code words are of fixed length.

3. (Previously Presented) Encoder according to claim 1, characterized in that the processing module includes:

- a function to calculate the current multiplexing capacity of the first data flow (S_H), based on the encoding module, and

- a function to extract a multiplexed part from the second data flow (S_L), determined on the basis of the current multiplexing capacity, to be carried by said redundant words.

4. (Previously Presented) Encoder according to claim 1, characterized in that it includes a transformation of a binary flow into a multi-valued variable flow.

5. (Original) Encoder according to claim 4, characterized in that it includes a transformation of a binary flow into a multi-valued variable flow, in particular using the transformations described in Table C.

6. (Original) Encoder according to claim 5, characterized in that it includes a transformation of a binary flow into a multi-valued variable flow, in particular using a generalized Euclidian decomposition based on a global variable given by the relationship (E9).

7. (Previously Presented) Encoder according to claim 1, characterized in that the encoding module includes an encoding table and in that the processing module includes:

- a function to read a multiplexing capacity of each current symbol of the first data flow (S_H) based on the encoding table and

- a function to extract a part of the second data flow (S_L) determined from the multiplexing capacity, to be carried by said redundant words.

8. (Original) Encoder according to claim 7, characterized in that the encoding table includes, for each symbol, an associated number of code words equal to a power of 2.

9. (Previously Presented) Encoder according to claim 1, characterized in that the encoding module includes a binary encoding tree containing, for each symbol in the first data flow, a first code word part, of variable length and shorter than a maximum length, and in that the processing module includes:

- a function to compute the multiplexing capacity for each current symbol of the first data flow (S_H) based on the first code word part of each symbol,

- a function to extract a part of the second data flow (S_L) determined from the multiplexing capacity, to be carried by said redundant words.

10. (Original) Encoder according to claim 9, characterized in that each symbol comprises a sequence of symbols.

11. (Previously Presented) Encoder according to claim 1, characterized in that each symbol comprises a sequence of symbols, in that the encoding module includes an arithmetic encoder designed to calculate, for a symbol sequence in the first data flow, a first code word part of variable length and shorter than a maximum length, and in that the processing module includes:

- a function to calculate the multiplexing capacity for each current symbol of the first data flow (S_H) based on the first code word part of each symbol,

- a function to extract a part of the second data flow (S_L) determined from the multiplexing capacity for each symbol, to be carried by said redundant words.

12. (Previously Presented) Encoder according to claim 9, characterized in that said part of the second data flow is concatenated with the first code word part up to the maximum length of the code word.

13. (Previously Presented) Encoder according to claim 1, characterized in that the second data flow is pre-encoded.

14. (Previously Presented) Encoder according to claim 1, characterized in that the rest of the second data flow is concatenated with the transmitted data.

15. (Previously Presented) Decoder designed to perform the inverse operations relative to those of the encoder according to claim 1.

16. (Original) Digital data compression method, characterized by the following steps:

a. establishing a match between symbols of the first data flow and code words, wherein, for certain symbols, there exist several words, termed redundant, corresponding to the same symbol, and

b. encoding the symbols of a first data flow based on the match obtained at step a., by selecting among the redundant words, on the basis of at least part of a second data flow.

17. (Previously Presented) Digital data compression method, characterized by:

a. establishing a match between symbols of the first data flow and code words, wherein, for certain symbols, there exist several words, termed redundant, corresponding to the same symbol, and

b. encoding the symbols of a first data flow based on the match obtained at step a., by selecting among the redundant words, on the basis of at least part of a second data flow, characterized by sub-functions according to claim 1.

18. (Previously Presented) Digital data decompression method, characterized by steps reciprocal to those of the method according to claim 16.